

CLAIMS

What is claimed is:

- 1 1. A method comprising:
2 recording a first last use of a first canonical register in a block of code after a
3 renaming, the first canonical register being mapped to a first original register; and
4 applying one of a first rollback and a first recovery to the first original register
5 based on whether the recorded first last use occurs before a first last definition of the first
6 original register in the block of code.
- 1 2. The method of claim 1 wherein applying one of the first rollback and the
2 first recovery comprises:
3 applying the first rollback to the first original register if the recorded first last use
4 occurs before the first last definition of the first original register; and
5 applying the first recovery to the first original register if the recorded first last use
6 does not occur before the first last definition of the first original register.
- 1 3. The method of claim 2 wherein applying the first rollback comprises:
2 replacing a first reference to a first target register with the first canonical register
3 when the first reference is a destination of a first last write to the first target register, the
4 first target register corresponding to the first original register after the renaming; and
5 replacing a second reference to the first target register with the first canonical
6 register when the second reference is a source of a first operation after the first last write to
7 the first target register.
- 1 4. The method of claim 2 wherein applying the first recovery comprises:
2 copying the first target register to the first canonical register at end of the block.
- 1 5. The method of claim 4 wherein copying the first target register comprises:
2 copying the first target register to a first unused temporary register; and
3 copying the first unused temporary register to the first canonical register.

1 6. The method of claim 5 further comprises:
2 recording a second last use of a second canonical register in the block of code after
3 the renaming, the second canonical register being mapped to a second original register; and
4 applying one of a second rollback and a second recovery to the second original
5 register based on whether the recorded second last use of the second canonical register
6 occurs before a second last definition of the second original register in the block of code.

1 7. The method of claim 6 wherein applying one of the second rollback and the
2 second recovery comprises:
3 applying the second rollback to the second original register if the recorded second
4 last use occurs before the second last definition of the second original register; and
5 applying the second recovery to the second original register if the recorded second
6 last use does not occur before the second last definition of the second original register.

1 8. The method of claim 7 wherein applying the second rollback comprises:
2 replacing a third reference to a second target register with the second canonical
3 register when the third reference is a destination of a second last write to the second target
4 register, the second target register corresponding to the second original register after the
5 renaming; and
6 replacing a fourth reference to the second target register with the second canonical
7 register when the fourth reference is a source of a second operation after the second last
8 write to the second target register.

1 9. The method of claim 8 wherein applying the second recovery comprises:
2 copying the second target register to the second canonical register at end of the
3 block.

1 10. The method of claim 9 wherein copying the second target register
2 comprises:
3 copying the second target register to a second unused temporary register before
4 copying the first unused temporary register to the first canonical register; and
5 copying the second unused temporary register to the second canonical register.

1 11. A computer program product comprising:
2 a machine useable medium having program code embedded therein, the program
3 code comprising:
4 computer readable program code to record a first last use of a first canonical
5 register in a block of code after a renaming, the first canonical register being mapped to a
6 first original register; and
7 computer readable program code to apply one of a first rollback and a first
8 recovery to the first original register based on whether the recorded first last use occurs
9 before a first last definition of the first original register in the block of code.

1 12. The computer program product of claim 11 wherein the computer readable
2 program code to apply one of the first rollback and the first recovery comprises:
3 computer readable program code to apply the first rollback to the first original
4 register if the recorded first last use occurs before the first last definition of the first
5 original register; and
6 computer readable program code to apply the first recovery to the first original
7 register if the recorded first last use does not occur before the first last definition of the first
8 original register.

1 13. The computer program product of claim 12 wherein the computer readable
2 program code to apply the first rollback comprises:
3 computer readable program code to replace a first reference to a first target register
4 with the first canonical register when the first reference is a destination of a first last write
5 to the first target register, the first target register corresponding to the first original register
6 after the renaming; and
7 computer readable program code to replace a second reference to the first target
8 register with the first canonical register when the second reference is a source of a first
9 operation after the first last write to the first target register.

1 14. The computer program product of claim 12 wherein the computer readable
2 program code to apply the first recovery comprises:

3 computer readable program code to copy the first target register to the first
4 canonical register at end of the block.

1 15. The computer program product of claim 14 wherein the computer readable
2 program code to copy the first target register comprises:

3 computer readable program code to copy the first target register to a first unused
4 temporary register; and

5 computer readable program code to copy the first unused temporary register to the
6 first canonical register.

1 16. The computer program product of claim 15 further comprises:

2 computer readable program code to record a second last use of a second canonical
3 register in the block of code after the renaming, the second canonical register being
4 mapped to a second original register; and

5 computer readable program code to apply one of a second rollback and a second
6 recovery to the second original register based on whether the recorded second last use of
7 the second canonical register occurs before a second last definition of the second original
8 register in the block of code.

1 17. The computer program product of claim 16 wherein the computer readable
2 program code to apply one of the second rollback and the second recovery comprises:

3 computer readable program code to apply the second rollback to the second original
4 register if the recorded second last use occurs before the second last definition of the
5 second original register; and

6 computer readable program code to apply the second recovery to the second
7 original register if the recorded second last use does not occur before the second last
8 definition of the second original register.

1 18. The computer program product of claim 17 wherein the computer readable
2 program code to apply the second rollback comprises:

3 computer readable program code to replace a third reference to a second target
4 register with the second canonical register when the third reference is a destination of a

5 second last write to the second target register, the second target register corresponding to
 6 the second original register after the renaming; and
 7 computer readable program code to replace a fourth reference to the second target
 8 register with the second canonical register when the fourth reference is a source of a
 9 second operation after the second last write to the second target register.

1 19. The computer program product of claim 18 wherein the computer readable
 2 program code to apply the second recovery comprises:
 3 computer readable program code to copy the second target register to the second
 4 canonical register at end of the block.

1 20. The computer program product of claim 19 wherein the computer readable
 2 program code to copy the second target register comprises:
 3 computer readable program code to copy the second target register to a second
 4 unused temporary register before copying the first unused temporary register to the first
 5 canonical register; and
 6 computer readable program code to copy the second unused temporary register to
 7 the second canonical register.

1 21. A system comprising:
 2 a processor; and
 3 a memory coupled to the processor to store program code, the program code, when
 4 executed, causing the processor to:
 5 record a first last use of a first canonical register in a block of code after a
 6 renaming, the first canonical register being mapped to a first original register; and
 7 apply one of a first rollback and a first recovery to the first original register
 8 based on whether the recorded first last use occurs before a first last definition of the first
 9 original register in the block of code.

1 22. The system of claim 21 wherein the program code causing the processor to
 2 apply one of the first rollback and the first recovery causes the processor to:
 3 apply the first rollback to the first original register if the recorded first last use
 4 occurs before the first last definition of the first original register; and

5 apply the first recovery to the first original register if the recorded first last use does
6 not occur before the first last definition of the first original register.

1 23. The system of claim 22 wherein the program code causing the processor to
2 apply the first rollback causes the processor to:
3 replace a first reference to a first target register with the first canonical register
4 when the first reference is a destination of a first last write to the first target register, the
5 first target register corresponding to the first original register after the renaming; and
6 replace a second reference to the first target register with the first canonical register
7 when the second reference is a source of a first operation after the first last write to the first
8 target register.

1 24. The system of claim 22 wherein the program code causing the processor to
2 apply the first recovery causes the processor to:
3 copy the first target register to the first canonical register at end of the block.

1 25. The system of claim 24 wherein the program code causing the processor to
2 copy the first target register causes the processor to:
3 copy the first target register to a first unused temporary register; and
4 copy the first unused temporary register to the first canonical register.

1 26. The system of claim 25 wherein the program code further causes the
2 processor to:
3 record a second last use of a second canonical register in the block of code after the
4 renaming, the second canonical register being mapped to a second original register; and
5 apply one of a second rollback and a second recovery to the second original register
6 based on whether the recorded second last use of the second canonical register occurs
7 before a second last definition of the second original register in the block of code.

1 27. The system of claim 26 wherein the program code causing the processor to
2 apply one of the second rollback and the second recovery causes the processor to:
3 apply the second rollback to the second original register if the recorded second last
4 use occurs before the second last definition of the second original register; and

5 apply the second recovery to the second original register if the recorded second last
6 use does not occur before the second last definition of the second original register.

1 28. The system of claim 27 wherein the program code causing the processor to
2 apply the second rollback causes the processor to:

3 replace a third reference to a second target register with the second canonical
4 register when the third reference is a destination of a second last write to the second target
5 register, the second target register corresponding to the second original register after the
6 renaming; and

7 replace a fourth reference to the second target register with the second canonical
8 register when the fourth reference is a source of a second operation after the second last
9 write to the second target register.

1 29. The system of claim 28 wherein the program code causing the processor to
2 apply the second recovery causes the processor to:

3 copy the second target register to the second canonical register at end of the block.

1 30. The system of claim 29 wherein the program code causing the processor to
2 copy the second target register causes the processor to:

3 copy the second target register to a second unused temporary register before

4 copying the first unused temporary register to the first canonical register; and

5 copy the second unused temporary register to the second canonical register.